

DATA TRANSMITTER

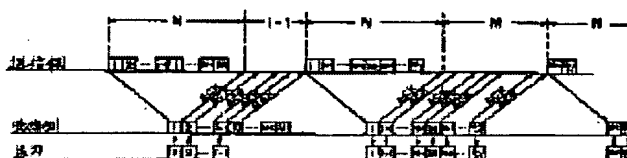
Patent number: JP7123079
Publication date: 1995-05-12
Inventor: TANAKA HIROKAZU
Applicant: TOKYO SHIBAURA ELECTRIC CO
Classification:
- international: H04L1/16; H04L29/08
- european:
Application number: JP19930301826 19931201
Priority number(s): JP19930301826 19931201; JP19930217599 19930901

Report a data error here

Abstract of JP7123079

PURPOSE: To decrease a buffer size and power consumption of a terminal equipment by keeping a throughput with a comparatively simple protocol in a system in which a loopback delay time is long.

CONSTITUTION: M-sets of consecutive blocks are sent continuously as one group in one transmission, a receiver side checks an error sequentially from the M blocks and returns an acknowledge ACK or a re-transmission request NAK. Suppose that an error is detected in an i-th block, blocks after the i-th block till the M-th block are aborted and the request NAK relating to the i-th block is returned to the sender side. The sender side receiving the NAK sends continuously the M blocks from the i-th block up to the (M+i-1)th block. The receiver side checks an error sequentially to the sent blocks the same as the 1st transmission and returns the ACK or the NAK. The processing is continued till all blocks are correctly received.



Data supplied from the **esp@cenet** database - Worldwide

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平7-123079

(43) 公開日 平成7年(1995)5月12日

(51) Int.Cl.⁶

H 0 4 L 1/16
29/08

識別記号

庁内整理番号

9371-5K

F I

技術表示箇所

9371-5K

H 0 4 L 13/ 00

3 0 7 Z

審査請求 未請求 請求項の数9 O L (全 19 頁)

(21) 出願番号 特願平5-301826

(22) 出願日 平成5年(1993)12月1日

(31) 優先権主張番号 特願平5-217599

(32) 優先日 平5(1993)9月1日

(33) 優先権主張国 日本 (J P)

(71) 出願人 000003078

株式会社東芝

神奈川県川崎市幸区堀川町72番地

(72) 発明者 田中 宏和

東京都日野市旭が丘3丁目1番地の1 株
式会社東芝日野工場内

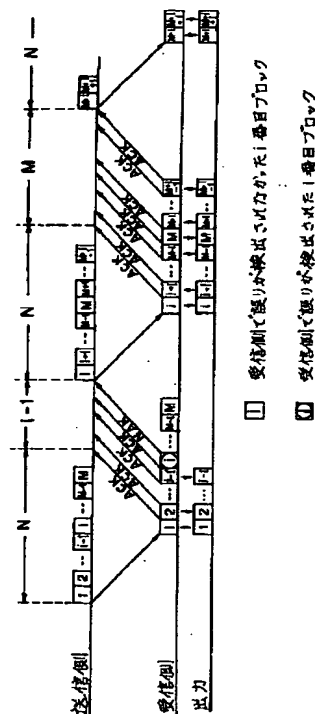
(74) 代理人 弁理士 木村 高久

(54) 【発明の名称】 データ伝送装置

(57) 【要約】

【目的】 折り返し遅延時間が長いシステムで、比較的簡単なプロトコルで、スループットを維持し、バッファ・サイズ、端末の消費電力を小さくすることを目的とする。

【構成】 1回の送信で連続するM個のブロックを1グループとして連続的に送信し、受信側でこのMブロックに順次誤り検出を行い、確認応答ACKまたは再送要求応答NAKを送り返す。i番目のブロックに誤りが検出されたとなると、i番目のブロック以降、M番目までのブロックを廃棄し、送信側にi番目のブロックに対する再送要求応答NAKを返す。再送要求応答NAKを受けた送信側では、i番目のブロックを先頭とするM+i-1番目までのMブロックを連続送信する。受信側ではこれに対して、1回目の場合と同じく順次誤り検出を行い、確認応答ACKまたは再送要求応答NAKを送り返す。このような処理を全ブロックが正しく受信されるまで続ける。



【特許請求の範囲】

【請求項 1】 送信側から送信されたデータが受信側で誤りと判断された場合に、送信側からデータの再送を行うデータ伝送装置において、

Nブロックのデータ信号の送信時間長に相当する折り返し遅延時間をもつ通信経路に対して、連続するMブロック ($M < N$) のデータ信号を送信する送信手段と、

前記送信手段により送信されたデータ信号を受信し、受信したM個の信号ブロックについてあらかじめ送信側で各信号ブロックに付加した誤り検出用のビットを用いて順次誤りを検出する検出手段と、

前記検出手段による誤り検出の結果、誤りが検出されなかった場合はそのブロックを利用者に出力すると同時に送信側に確認応答を送り返し、誤りが検出された場合にはそのブロックを出力しないで送信側に再送要求応答を送り返す応答手段と、

前記応答手段により送り返された前記応答を受信し、その応答が再送要求応答であった場合、誤りの検出された信号ブロックを含めて連続するM個の信号ブロックを再送信する再送信手段を具備するデータ伝送装置。

【請求項 2】 送信側から送信されたデータが受信側で誤りと判断された場合に、送信側からデータの再送を行うデータ伝送装置において、

受信側からの再送要求に応じて、ブロック化された送信すべきデータ、或いは送信すべきデータをもとに生成された誤り訂正用のパリティ・ビットからなるブロック化されたデータに対し、誤り検出用のビットを各信号ブロックに付加する処理を施した後、Nブロックのデータ信号の送信時間長に相当する折り返し遅延時間をもつ通信経路に対して、連続するMブロック ($M < N$) のデータ信号を送信する送信手段と、

前記送信手段により送信されたデータを受信し、受信したM個の信号ブロックについて、あらかじめ送信側で各信号ブロックに付加された前記誤り検出用のビットを用いて誤りを検出する検出手段と、

前記検出手段による誤り検出の結果、誤りが検出されなかった場合はそのブロックを利用者に出力すると同時に送信側に確認応答を送り返し、誤りが検出された場合にはそのブロックを出力しないで送信側に再送要求応答を送り返すと共に、誤りの検出された信号ブロックとそれに続くM-1個の信号ブロックを保持する保持手段と、前記再送された信号ブロックと前記保持手段により保持された信号ブロックとを用いて誤り訂正を行う訂正手段とからなるデータ伝送装置。

【請求項 3】 送信側から送信されたデータが受信側で誤りと判断された場合に、送信側からデータの再送を行うデータ伝送装置において、

受信側からの応答信号を待たずに常に連続して信号ブロックを送信し、受信側からの再送要求に応じて、ブロック化された送信すべきデータ、或いは送信すべきデータ

をもとに生成された誤り訂正用のパリティ・ビットからなるブロック化されたデータに対し、誤り検出用のビットを各信号ブロックに付加する処理を施した後、連続する一定数K個の信号ブロックを送信する送信手段と、前記送信手段により送信されたデータを受信し、受信したK個の信号ブロックについて、あらかじめ送信側で各信号ブロックに付加された前記誤り検出用のビットを用いて誤りを検出する検出手段と、

前記検出手段による誤り検出の結果、誤りが検出されなかった場合はそのブロックを利用者に出力すると同時に送信側に確認応答を送り返し、誤りが検出された場合にはそのブロックを出力しないで送信側に再送要求応答を送り返すと共に、誤りの検出された信号ブロックとそれに続くK-1個の信号ブロックを保持する保持手段と、前記再送された信号ブロックと前記保持手段により保持された信号ブロックとを用いて誤り訂正を行う訂正手段とからなるデータ伝送装置。

【請求項 4】 送信側から送信されたデータが受信側で誤りと判断された場合に、送信側からデータの再送を行うデータ伝送装置において、

Nブロックのデータ信号の送信時間長に相当する折り返し遅延時間をもつ通信経路に対して、連続するkNブロック (kは自然数) のデータ信号を送信する送信手段と、

前記送信手段により送信されたデータ信号を受信し、受信したkN個の信号ブロックについてあらかじめ送信側で各信号ブロックに付加した誤り検出用のビットを用いて順次誤りを検出する検出手段と、

前記検出手段による誤り検出の結果、誤りが検出されなかった場合はそのブロックを利用者に出力すると同時に送信側に確認応答を送り返し、誤りが検出された場合にはそのブロックを出力しないで送信側に再送要求応答を送り返す応答手段と、

前記応答手段により送り返された前記応答を受信し、その応答が再送要求応答であった場合、誤りの検出された信号ブロックを直ぐに再送信する再送信手段とを具備し、

前記kN個の信号ブロックの全てについて受信側からの応答が確認応答になるまで前記再送信手段による再送信を行い、前記kN個の信号ブロックが全部送信完了した後、前記送信手段が新たに連続するkNブロックのデータ信号を送信する事を特徴とするデータ伝送装置。

【請求項 5】 送信側から送信されたデータが受信側で誤りと判断された場合に、送信側からデータの再送を行うデータ伝送装置において、

Nブロックのデータ信号の送信時間長に相当する折り返し遅延時間をもつ通信経路に対して、連続するkNブロック (kは自然数) のデータ信号を送信する送信手段と、

前記送信手段により送信されたデータ信号を受信し、受

信した kN 個の信号ブロックについてあらかじめ送信側で各信号ブロックに付加した誤り検出用のビットを用いて順次誤りを検出する検出手段と、

前記検出手段による誤り検出の結果、誤りが検出されなかった場合はそのブロックを利用者に出力すると同時に送信側に確認応答を送り返し、誤りが検出された場合にはそのブロックを出力しないで送信側に再送要求応答を送り返す応答手段と、

前記応答手段により送り返された前記応答を受信し、その応答が再送要求応答であった場合、誤りの検出された信号ブロックを直ぐに再送信し、該再送信するブロックと次に送信するブロックの間に送信の空白時間がある場合、該空白時間に前記再送信ブロックを連続して送信する再送信手段とを具備し、

前記 kN 個の信号ブロックの全てについて受信側からの応答が確認応答になるまで前記再送信手段による再送信を行い、前記 kN 個の信号ブロックが全部送信完了した後、前記送信手段が新たに連続する kN ブロックのデータ信号を送信する事を特徴とするデータ伝送装置。

【請求項 6】 前記訂正手段が、

送信すべきデータ 1 ブロックと、それをもとに生成された誤り訂正用のパリティ・ビットから成るブロックから構成される誤り訂正符号で、1 回の再送要求に対しを誤り訂正用のパリティ・ビットから成るブロックを 1 ブロックずつ送信し、送信された信号ブロックは前の信号ブロックに次々と、最大 $L-1$ ブロックまで付加され、付加される度に訂正符号のより高い符号となりうる符号を用いて実現することを特徴とする請求項 2 または 3 記載のデータ伝送装置。

【請求項 7】 前記送信すべきデータをもとに生成された誤り訂正用のパリティ・ビットとして、それ自身からもとの情報を復元することができるパリティ・ビットを用いることを特徴とする請求項 2 乃至 5 のいずれかに記載のデータ伝送装置。

【請求項 8】 前記送信手段が、

前記通信経路の通信状態に応じて前記自然数 k を変えることで、1 回に送信するブロックの数を適応的に変化することができることを特徴とする請求項 4 または 5 記載のデータ伝送装置。

【請求項 9】 送信側又は受信側にブロック伝送の誤り率の推定又は測定を行う計測装置を更に設け、該計測装置による誤り率の推定又は測定結果が、予め決められた規定値よりも低い値であるときは、前記再送信手段が、前記再送要求応答を受けても、前記確認応答として処理することを特徴とする請求項 1 乃至 8 のいずれかに記載のデータ伝送装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、デジタル・データ通信などに用いられるデータ伝送装置に関し、ことに高いス

ループットを示し、かつアルゴリズムを簡略化したデータ伝送装置に関する。

【0002】

【従来の技術】 デジタル・データ通信など高い信頼性が要求されるシステムにおいては、従来 ARQ (自動再送要求) 方式による誤り制御が広く用いられている。ARQ 方式はその再送手順により次の 3 つの基本方式に分類できる。

【0003】 (1) Stop-And-Wait (SAW) ARQ 方式

信号ブロックが送信されると、送信側では受信側からの返事を待ち、ACK (肯定応答、ここでは確認応答) ならば次ぎのブロックを送信し、NAK (否定応答、ここでは再送要求応答) ならば同じブロックを再送する。

【0004】 (2) Go-Back-N (GBN) ARQ 方式

折り返し遅延時間 (RTD) 中にも連続的にブロックを送信し続け、ACK または NAK が返ってきた時点で、次のブロックを送信するか、前の連続したブロックを再送するかを判定する。ここで、 N は RTD 中に送信できるブロックの数を表す。この方式は通信路の状態がよく、RTD が短い場合は非常に効率が良くなるが、通信路の状態が悪く、RTD が長い場合には極端に効率が悪化する。

【0005】 (3) Selective-Repeat (SR) ARQ 方式

折り返し遅延時間 (RTD) 中にも連続的にブロックを送信し続け、NAK が返ってきたブロックのみを再送する。そのため、誤りのあるブロックの後に受信された正しいブロックを保存するためのバッファを有し、再送された信号が ACK になったとき、バッファ内に記録されているブロックと共に送信された順序でユーザに出力する。この方式はこの 3 つの中で最も効率の良い方法であるが、論理が複雑になり、受信側に膨大な容量のバッファが必要になる。

【0006】 ARQ 方式に関しては例えば IEEE COMMUNICATIONS MAGAZINE の第 22 巻 第 12 号の S. Lin, D. J. Costello, Jr., M. J. Miller 著の論文 "Automatic-Repeat-Request Error-Control Schemes" に記されている。

【0007】 一般に、移動体通信においては、電波が多重経路を経て受信されるマルチパスフェージングを受けるため、誤り率の変動が大きい。このような通信経路上を高速のデジタル・データが伝送されると、誤りの非常に多いブロックが長い区間連続して受信される場合から、誤りのないブロックが長い区間連続して受信される場合まで、受信ブロックの誤り方は非常に幅広い範囲に亘る。従って適用する ARQ 方式もそのときの通信路の

状態に合わせて効率よく動作する方式が望まれる。そのような方式として幾つかの方法が工夫されて発表されている。その1つとして、Type IIハイブリッドARQ方式がある。Type IIハイブリッドARQ方式は、ARQ方式に誤り訂正方式を組み合わせた方式の1つで、最初は情報ビットに誤り検出用のパリティ・ビットのみを付加して送信し、NAKが返ってきたら誤り訂正用のパリティ・ビットを再送する。この再送されるパリティ・ビットは、そこから情報ビットが取り出せるように構成されており、誤りが検出されなければそこから情報ビットを再生し、誤りが検出されたならばバッファに保存された情報ビットと合わせて誤り訂正を行う。

【0008】R. A. Comroe, D. J. Costelloらは誤り訂正符号にブロック符号を、伝送手順にARQ方式のSelective-Repeat (SR) モードを用いたType IIハイブリッドARQ方式の移動体通信システムへの適用を提案している。(例えばIEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONSの第SAC-2巻 第4号のR. A. Comroe, D. J. Costello著の論文“ARQ Schemes for Data Transmission in Mobile Radio Systems”を参照)。

【0009】この場合の伝送手順及び処理手順を図14、図15(a)～(g)で説明する。図15(a)～(g)中11～14は処理を、20、21は判断を示す。

【0010】図14はSRモードにもとづくブロックの伝送手順を示す。図14でブロック1～3及び5は1回目の送信で誤りが検出できなかったためブロック1～3はそのまま、ブロック5は図15(b)に示すように、ブロック4の処理後ユーザに出力される。同時にこれらに対するACKが受信側から送信側に返される。ブロック4、6は誤りが検出されたので、これらのブロックは一旦バッファに保存され、(図15(a)、図15(d)に示す。)同時にこれらに対するNAKが受信側から送信側に返される。NAKを受けた送信側は2回目の送信でブロック4、6の再送ブロック4', 6'を送信する。

【0011】ここで再送ブロック4', 6'はそれぞれ1回目に送信されたブロック4、6を情報ビットとする誤り訂正符号のパリティ・ビットからなる。これらの再生ブロックは図15(c)、図15(e)、図15(f)に示す手順により処理される。誤りが検出されなかった再送ブロック6'は、パリティ・インバータにより元の情報ビット(ブロック6)が復元され、1回目の送信でバッファに保存されたブロック6は廃棄される。一方、誤りが検出された再送ブロック4'は1回目の送信でバッファに保存されたブロック4と合わせて誤り訂

正を行う。以上の処理の結果、それでもなお誤りがある場合には1回目の送信でバッファに保存されたブロックは廃棄され、代わりに2回目に送信された再送ブロックがバッファに保存される。受信側では再送ブロックの誤り検出を行い、誤りがあればバッファに保存された2回目の再送ブロックと併せて再び誤り訂正を行う。以下同じ要領で処理が繰り返される。

【0012】しかしながらこの方式は、受信側で常に連続した順番でブロックを受信するわけではないので、アルゴリズムが非常に複雑になると言う問題がある。

【0013】また、誤り率の変動が大きいマルチパス・フェージング通信路に高速のデジタル・データが伝送されると、誤りの非常に多いブロックが長い区間連続して受信されることがあり、この場合でも、再送されたブロックがACKになるまで正しく受信したデータを保存しておくバッファが、オーバーフローを起こす可能性が高く、バッファ容量が膨大になると言う問題がある。

【0014】また別な解決方法として、A. R. K. Sastry, M. Moeneclaey, F. Argentiらによって提案されている一連の方法がある。

【0015】A. R. K. Sastryの方法はNAKを受信すれば誤ったブロックをp回連続して繰り返し、p回とも正しくなかったときにのみ、そのブロックをもう一度同じ手順で再送する方式である。この方式は例えばIEEE TRANSACTIONS ON COMMUNICATIONSの第COM-23巻 第4号のA. R. K. Sastry著の論文“Improving Automatic Repeat-Request (ARQ) Performance on Satellite Channels Under High Error Condition”に記されている。

【0016】この方法を更に一般化し、pの値を通信路の状態に応じて最適化する方法がM. Moeneclaeyらによって提案されている。

【0017】この方法で、pが通信路の状態から2になっているとすると、1度NAKが返ると同じブロックを2回再送し、2回ともNAKが返されると、今度は同じブロックを3回連続して送信するようにして、通信路の状態に応じた最適なKを選択する。この方式は、例えばIEEE TRANSACTIONS ON COMMUNICATIONSの第COM-34巻 第2号のM. Moeneclaey他著の論文“Throughput Optimization for a Generalized Stop-and-Wait ARQ Scheme”に記されている。

【0018】さらにF. Argenti等は、幾つかの連続するブロックを1グループとしてグループ内のブロックを連続的に送信し、受信側からのACK/NAKを待ち、NAKを受信すると、そのグループのブロックの

内、誤ったブロックから後ろのブロックを再送する。これにより再送のための遅延時間が短縮され、スループットが改善できる。この方式に関しては、例えば IEEE ELECTRONICS LETTERS 第28巻第9号のF. Argenti, G. Benelli, A. Garzelli 著の論文 "Generalized Stop-and-Wait Protocol" に記されている。

【0019】F. Argentiらの方式は従来のSWAプロトコルに比べて通信路の状態の良いところではかなりのスループットの向上が期待できる。しかしこの方式の2回目以降の送信は、誤りのあったブロックからM番目のブロックまでしかないため、1回目の送信に比べて効率は良くない。

【0020】一方、送信すべきデータ1ブロックと、それをもとに生成された誤り訂正用のパリティ・ビットから成るブロックで構成される誤り訂正符号で、1回の再送要求に対しを誤り訂正用のパリティ・ビットから成るブロックを1ブロックずつ送信し、送信された信号ブロックは前の信号ブロックに次々と付加され、付加される度に訂正符号のより高い符号となりうる符号を用いた方法が、例えば、H. Krishna and S. Morgera; "A new error control scheme for hybrid ARQ system", IEEE Trans. Commun., vol. COM-35, no. 10, pp. 981-990, Oct. 1987等に提案されている。

【0021】また、送信すべきデータをもとに生成された誤り訂正用のパリティ・ビットで、それ自身からものと信号を復元できる符号を用いた方法が、例えば、S. Lin and P. S. Yu; "A hybrid ARQ scheme with parity retransmission for error control of satellite channels", IEEE Trans. Commun., vol. COM-30, no. 7, pp. 1701-1719, Jul. 1982等に示されている。

【0022】この様な符号を装置に使用することにより、誤り訂正の効率を改善できる可能性は大きい。

【0023】また、更に別な解決方法として、M. J. Miller等は、SRプロトコルとGBNプロトコルとを組み合わせることで、受信バッファのオーバーフローを防ぐ方法を M. J. Miller and others: "The Analysis of Some Selective-repeat ARQ with Finite Receiver Buffer", IEEE Trans. Commun., vol. COM-29, no. 4 に提案している。

【0024】この方式は、SRモードのARQ方式で動作しているときに、受信側で最初に誤りがあると判断さ

れたブロックが ν 回の再送に対して全てNAKと送信側で判断された場合は、モードをSRモードからGBNモードのARQ方式に切り替えて送信を行う。そうして送信側で最初に誤りがあると判断されたブロックがACKと判断されれば再びSRモードに戻って伝送を行う。

【0025】図16に $N=5$ 、 $\nu=1$ の例を示す。この例を図に添って具体的に説明すると、最初、受信側で誤りと判断されたブロック2、ブロック4、ブロック7について受信側で送信側に対してNAKを返し、送信側は先ずSRモードで再送を行う。その結果、ブロック2及びブロック4は今度は受信側で誤りがないと判断され、ACKが送信側に返される。一方、ブロック7は受信側で再び誤りと判断されたので、受信側は送信側にNAKを返す。このNAKを受けた送信側では、モードをSRモードからGBNモードに切り替えて送信を行う。すなわち、再送ブロック7と、それに続く4つのブロックをブロック7に対してACKが返るまでGBNプロトコルで伝送を行う。そうして送信側でブロック7に対してACKを受けたと判断されると、再度SRモードに戻って、その後のブロックの伝送を行う。このM. J. Miller等の論文には、更にも一つの手段として、SRプロトコルと、誤ったブロックをACKが返るまで連続して送信し続けるStutter (ST) モードの組み合わせ方を提案している。この方式は先ずSRモードで動作していて、このとき最初に誤りがあると判断されたブロックが ν 回の再送に対して全てNAKと送信側で判断された場合に、モードをSRモードからSTモードに切り替えて送信を行う。そうしてこの誤りと判断されたブロックを連続的に送信し、ACKが送信側で受信されれば再びSRモードに戻って次の伝送を行う。

【0026】この例を、 $N=4$ 、 $\nu=1$ の場合について図17に示す。この例を図に添って説明すると、受信側で誤りと判断されたブロック5及びブロック7は、受信側から送信側にNAKが返され、送信側はSRモードでもう一度再送を行う。しかしブロック5はまたNAK出会ったため送信側はSRモードからSTモードに切り替えてブロック5に対してACKが返るまで連続して再送を行う。そうして送信側でACKを受けたことが判断されると、再びSRモードに戻ってその後の伝送を行う。

【0027】これらのM. J. Miller等が提唱する方法は、受信バッファのオーバーフローは防ぐことができるが、複数の論理を切り替えるため、アルゴリズムが複雑になる欠点がある。

【0028】

【解決しようとする課題】上述したように従来のARQ方式においては、通信状態の悪いところでは、受信バッファの容量が大きくなり、スループットが急激に悪くなるなどの現象がみられ、受信バッファのオーバーフローを防ごうとすると、ややもするとアルゴリズムが複雑になる事が多かった。

【0029】そこでこの発明では、これらの問題を改良して、小形携帯端末を用いてデジタル・データ通信を衛星通信システムなどのRTDが長いシステムで行えるようにすることを目標に、プロトコルをできるだけ簡単に、バッファはできるだけ小さく、更に端末の消費電力はできるだけ小さくなるようにシステムを構築して、装置の小形化を図り、しかもスループットをできるだけ高くするようにすることを目的とする。

【0030】

【課題を解決するための手段】上記目的を達成するため、送信側から送信されたデータが受信側で誤りと判断された場合に、送信側からデータの再送を行うデータ伝送装置を、Nブロックのデータ信号の送信時間長に相当する折り返し遅延時間をもつ通信経路に対して、連続するMブロック ($M < N$) のデータ信号を送信する送信手段と、この送信手段により送信されたデータ信号を受信し、受信したM個の信号ブロックについてあらかじめ送信側で各信号ブロックに付加した誤り検出用のビットを用いて順次誤りを検出する検出手段と、この検出手段による誤り検出の結果、誤りが検出されなかった場合はそのブロックを利用者に出力すると同時に送信側に確認応答を送り返し、誤りが検出された場合にはそのブロックを出力しないで送信側に再送要求応答を送り返す応答手段と、応答手段により送り返された応答を受信し、その応答が再送要求応答であった場合、誤りの検出された信号ブロックを含めて連続するM個の信号ブロックを再送信する再送信手段で構成する。

【0031】また、前記データ伝送装置を、受信側からの再送要求に応じて、ブロック化された送信すべきデータ、或いは送信すべきデータをもとに生成された誤り訂正用のパリティ・ビットからなるブロック化されたデータに対し、誤り検出用のビットを各信号ブロックに付加する処理を施した後、Nブロックのデータ信号の送信時間長に相当する折り返し遅延時間をもつ通信経路に対して、連続するMブロック ($M < N$) のデータ信号を送信する送信手段と、送信手段により送信されたデータを受信し、受信したM個の信号ブロックについて、あらかじめ送信側で各信号ブロックに付加された前記誤り検出用のビットを用いて誤りを検出する検出手段と、検出手段による誤り検出の結果、誤りが検出されなかった場合はそのブロックを利用者に出力すると同時に送信側に確認応答を送り返し、誤りが検出された場合にはそのブロックを出力しないで送信側に再送要求応答を送り返すと共に、誤りの検出された信号ブロックとそれに続くM-1個の信号ブロックを保持する保持手段と、再送された信号ブロックと保持手段により保持された信号ブロックとを用いて誤り訂正を行う訂正手段とで構成する。

【0032】また、前記データ伝送装置を、受信側からの応答信号を待たずに常に連続して信号ブロックを送信し、受信側からの再送要求に応じて、ブロック化された

送信すべきデータ、或いは送信すべきデータをもとに生成された誤り訂正用のパリティ・ビットからなるブロック化されたデータに対し、誤り検出用のビットを各信号ブロックに付加する処理を施した後、連続する一定数K個の信号ブロックを送信する送信手段と、送信手段により送信されたデータを受信し、受信したK個の信号ブロックについて、あらかじめ送信側で各信号ブロックに付加された誤り検出用のビットを用いて誤りを検出する検出手段と、検出手段による誤り検出の結果、誤りが検出されなかった場合はそのブロックを利用者に出力すると同時に送信側に確認応答を送り返し、誤りが検出された場合にはそのブロックを出力しないで送信側に再送要求応答を送り返すと共に、誤りの検出された信号ブロックとそれに続くK-1個の信号ブロックを保持する保持手段と、再送された信号ブロックと保持手段により保持された信号ブロックとを用いて誤り訂正を行う訂正手段とで構成する。

【0033】また、前記データ伝送装置を、Nブロックのデータ信号の送信時間長に相当する折り返し遅延時間をもつ通信経路に対して、連続するkNブロック (kは自然数) のデータ信号を送信する送信手段と、送信手段により送信されたデータ信号を受信し、受信したkN個の信号ブロックについてあらかじめ送信側で各信号ブロックに付加した誤り検出用のビットを用いて順次誤りを検出する検出手段と、検出手段による誤り検出の結果、誤りが検出されなかった場合はそのブロックを利用者に出力すると同時に送信側に確認応答を送り返し、誤りが検出された場合にはそのブロックを出力しないで送信側に再送要求応答を送り返す応答手段と、応答手段により送り返された応答を受信し、その応答が再送要求応答であった場合、誤りの検出された信号ブロックを直ぐに再送信する再送信手段とで構成し、kN個の信号ブロックの全てについて受信側からの応答が確認応答になるまで再送信手段による再送信を行い、kN個の信号ブロックが全部送信完了した後、送信手段が新たに連続するkNブロックのデータ信号を送信するようにする。

【0034】また、前記データ伝送装置を、Nブロックのデータ信号の送信時間長に相当する折り返し遅延時間をもつ通信経路に対して、連続するkNブロック (kは自然数) のデータ信号を送信する送信手段と、送信手段により送信されたデータ信号を受信し、受信したkN個の信号ブロックについてあらかじめ送信側で各信号ブロックに付加した誤り検出用のビットを用いて順次誤りを検出する検出手段と、検出手段による誤り検出の結果、誤りが検出されなかった場合はそのブロックを利用者に出力すると同時に送信側に確認応答を送り返し、誤りが検出された場合にはそのブロックを出力しないで送信側に再送要求応答を送り返す応答手段と、応答手段により送り返された応答を受信し、その応答が再送要求応答であった場合、誤りの検出された信号ブロックを直ぐに再

送信し、再送信するブロックと次に送信するブロックの間に送信の空白時間がある場合、空白時間に再送信ブロックを連続して送信する再送信手段とで構成し、 kN 個の信号ブロックの全てについて受信側からの応答が確認応答になるまで再送信手段による再送信を行い、 kN 個の信号ブロックが全部送信完了した後、送信手段が新たに連続する kN ブロックのデータ信号を送信するようにする。

【0035】さらにここで、訂正手段を、送信すべきデータ1ブロックと、それをもとに生成された誤り訂正用のパリティ・ビットから成るブロックから構成される誤り訂正符号で、1回の再送要求に対しを誤り訂正用のパリティ・ビットから成るブロックを1ブロックずつ送信し、送信された信号ブロックは前の信号ブロックに次々と、最大 $L-1$ ブロックまで付加され、付加される度に訂正符号のより高い符号となりうる符号を用いて実現する。

【0036】また、送信すべきデータをもとに生成された誤り訂正用のパリティ・ビットとして、それ自身からもとの情報を復元できることを特徴とするパリティ・ビットを用いて実現する。

【0037】また送信手段を、通信経路の通信状態に応じて自然数 k を変えることで、1回に送信するブロックの数を適応的に変化することができるようにする。

【0038】さらに、送信側又は受信側にブロック伝送の誤り率の推定又は測定を行う計測装置を設け、計測装置による誤り率の推定又は測定結果が、予め決められた規定値よりも低い値であるときは、再送信手段が、再送要求応答を受けても、確認応答として処理するようにする。

【0039】

【作用】本発明のデータ伝送装置では、送信要求に応じて行われる2回目以後の送信時においても、1回目の送信時と同じ数の連続するブロックを送信するようにし、また、送信要求ごとにデータとパリティ・ビットが組み合わされて、訂正能力がより高くなる符号を用いるようにし、また、誤り訂正用のパリティ・ビットは、それ自身からもとの情報を復元できるものを用いるようにした。従って、簡単なプロトコル、比較的短いバッファを用いても、高いスループットを実現することができる。

【0040】

【実施例】以下図面に添って本発明の実施例を説明する。

【0041】図1は、本発明の一実施例の信号伝送手順で、これは、F. Argentiらの方式を改良したものを示す。

【0042】図1において折り返し遅延時間(RTD)中に N 個のブロックが送信できるとすると、1回の送信で連続する M 個のブロック($M < N$)を1グループとして、1回目の送信で最初の M 個のブロックを連続的に送

信し、受信側でこの M ブロックに順次誤り検出を行い、ACK/NAKを送り返す。

【0043】今、 i 番目のブロックに誤りが検出されたとすると、 i 番目のブロック以降、 M 番目までのブロックは廃棄され、送信側に i 番目のブロックに対するNAKが返される。NAKを受けた送信側では、もう一度 i 番目のブロックから M 番目のブロックとそれに続く $i+1$ 番目から $M+i-1$ 番目のブロックを送信する。受信側ではこれに対して、1回目の場合と同じく順次誤り検出を行い、ACK/NAKを送り返す。このような処理を繰り返して、以下、全てのブロックが受信されるまで続ける。

【0044】この実施例では、2回目以後の送信時においても、1回目の送信時と同じ数の連続するブロックを送信するので、高いスループットを維持したまま従来よりアルゴリズムを簡略化でき、バッファのオーバーフローを防ぎ、通信状態の悪いところで急激にスループットが劣化するのを防止できる。

【0045】また、更にこの方式に誤り訂正符号を効率よく組み合わせることで、通信路の状態が悪い場合のスループットの急激な劣化を防止する事ができる。

【0046】図2は、本発明の他の実施例の送信手順である。ここでは $N=5$ 、 $M=2$ の場合で説明する。また図3は、図2の各ブロックに対する処理の内容を示す。

【0047】図2の例では、受信側で誤りが検出されなかった場合は、ACKが返され、受信内容はそのままユーザに出力される。 i 番目のブロックで誤りが検出されると、受信側は送信側にNAKを返し、ブロック i 、 $(i+1)$ をバッファに保存する。NAKを受けた送信側は2回目の送信でブロック i 、 $(i+1)$ の再送ブロック i' 、 $(i+1)'$ を送信する。

【0048】ここで、再送ブロック i' 、 $(i+1)'$ はそれぞれ1回目に送信されたブロック i 、 $(i+1)$ を情報ビットとする誤り訂正符号のパリティ・ビットからなる。

【0049】つぎに、1回目の送信ブロック i と2回目の再送ブロック i' とを合わせて誤り訂正を行う。ここで誤り訂正符号は、パリティ・ビットを次々と追加することによって、より訂正能力の高い符号となりうる符号を用いる。

【0050】誤り訂正の結果、誤りが訂正出来た場合は、ACKが返される。誤りが訂正出来なかった場合は、送信側にNAKを返して、3回目の再送ブロック i'' を受信する。そしてブロック i 、 i' 、 i'' を合わせて誤り訂正を行う。このような操作を誤りが訂正できるまで、或いはパリティ・ビットからなるブロックが最大の $L-1$ 個付加されるまで繰り返される。誤りが訂正できた場合には、ブロック i をユーザに出力し、送信側にACKを返す。

【0051】一方、ブロック $(i+1)$ にも受信後、ブ

ブロック i と全く同じ処理が行われる。そしてブロック i よりも先に処理が終了した場合にはバッファに保存され、ブロック 3 の処理が終了後出力される。

【0052】図 3 (a) 及び (c) はブロック 3、図 3 (f) 及び (g) はブロック 5 について、1 回目の送信ブロックと 2 回目の再送ブロックとを合わせて誤り訂正が成功した場合を示す。また、図 3 (b)、(d) および (e) では、ブロック 4 について 3 回目の再送で誤りが訂正された場合を示している。

【0053】なお、図 3 (a) ~ (h) で、10~14 は処理を 20、21 は判断を示し、これ以後の図でも同じ表現を取る。

【0054】これにより、誤りのレベルに応じて必要最少限の訂正能力の誤り訂正符号を構成し、訂正することができる。またバッファの大きさは $M \cdot (L-1)$ ブロック分で良い。

【0055】図 4 に、本発明のさらに他の実施例の送信手順を示す。また図 5 は、図 4 の各タイミングにおける処理の内容を示す。

【0056】この例も図 2 の例と同様に $N=5$ 、 $M=2$ の場合で説明する。図 4 において、1 回目の送信で誤りが検出されなかった場合は ACK が返送され、受信内容がそのままユーザに出力される。誤りが検出された場合は、NAK が送信側に返され、その後、次のような手順で処理される。即ち、まず、ブロック i とそれに続くブロック $(i+1)$ がバッファに保存される。そして NAK を受けた送信側では 2 回目の送信でブロック i 、 $(i+1)$ を送信する。

【0057】ここで再送ブロック i 、 $(i+1)$ はそれぞれ 1 回目に送信されたブロック i 、 $(i+1)$ を情報ビットとする誤り訂正符号のパリティ・ビットからなる。これらの再送ブロックは、受信したブロック i に誤りが検出されなかった場合、パリティ・インバートによりブロック i が復元され、ユーザに出力される。

(この操作手順は図 5 (f) と図 5 (g) にブロック 5、5' の場合で示した。) また、受信したブロック i に誤りが検出された場合、図 5 (c) にブロック 3 で示したように、バッファに保存されたブロック i と合わせて誤り訂正がなされる。それでもなお誤りがあるときは、バッファに保存されたブロック i が廃棄され、代わりに 2 回目に送信されたブロック i が保存される。同時に NAK が送信側に返され、送信側から 3 回目の再送ブロックとしてブロック i 、 $(i+1)$ が送信される。受信側では再送されたブロック i の誤り検出を行い、誤りがなければそのままユーザに出力し、誤りがあればバッファに保存されたブロック i と合わせて誤り訂正を行う。(この操作手順は図 5 (d) と図 5 (e) にブロック 4、4' の場合で示した。) 以下同様の要領で処理が繰り返される。

【0058】一方ブロック $(i+1)$ には受信後、ブ

ロック i と全く同じ処理が行われる。そしてブロック i よりも先に処理が終了した場合にはバッファに保存され、ブロック i の処理が終了後、出力される。

【0059】この方式によれば、受信機側の持つバッファの大きさは M ブロック分で良い。また、2 回目以降の送信時においても、1 回目送信時と同じ数の連続するブロックを送信することで、効率を高めることができる。また、通信路の状態が悪化してもスループットが急激に劣化することなく、通信路の状態の悪いところでも効率良く伝送することができる。

【0060】図 6 は本発明の更に他の実施例を説明する図である。ここでは Go-Back-N (GBN) ARQ 方式に本発明を適用している。この場合は、折り返し遅延時間 (RTD) 中にも連続的にブロックを送信し続ける。RTD 中に送信できるブロック数を M として、 $M=2$ の場合について説明する。

【0061】図 6 で、受信側で誤りが検出されなかった場合は、ACK が返送され、受信内容はそのままユーザに出力される。ブロック i に誤りが検出された場合は、NAK を返し、ブロック i 、 $(i+1)$ をバッファに保存する。NAK を受けた送信側では 2 回目の送信でブロック i 、 $(i+1)$ の再送ブロック i' 、 $(i+1)'$ を送信する。ここで再送ブロック i' 、 $(i+1)'$ はそれぞれ 1 回目に送信されたブロック i 、 $(i+1)$ を情報ビットとする誤り訂正符号のパリティ・ビットからなる。

【0062】これらの再送ブロックの処理は 1 回目の送信ブロック i と 2 回目の送信ブロック i' とを合わせて誤りの訂正を行う。この間の処理の内容を図 7 (a) ~ 図 7 (f) にブロック i をブロック 3 として示した。

【0063】ここで誤り訂正符号は、送信すべきデータ 1 ブロックと、それをもとに生成された誤り訂正符号のパリティ・ビットからなる $L-1$ 個のブロックとで構成される符号で、1 回の再送要求に対してパリティ・ビットからなるブロックを 1 ブロックずつ送信し、送信されたブロックは前のブロックに次々に、最大 $L-1$ ブロックまで付加され、付加される度に訂正能力が高くなる符号である。

【0064】もし 1 回目の送信ブロック i と 2 回目の送信ブロック i' とを合わせて誤りの訂正を行っても誤りが訂正できなかった場合、送信側に NAK を返して 3 回目の再送ブロック i'' 、 $(i+1)''$ を受信する。そしてブロック i 、 i' 、 i'' を合わせて誤り訂正を行う。この操作は誤りが訂正できるまで、又はパリティ・ビットから成るブロックが最大 $L-1$ 個付加されるまで繰り返す。

【0065】誤り訂正ができた場合、ブロック i をユーザに出力し、送信側に ACK を返す。送信側は ACK を受けて新しいブロック $(i+2)$ を送信する。

【0066】この間、ブロック $(i+1)$ は受信後、ブ

ブロック i と全く同じ処理が行われる。そしてブロック i よりも先に処理が終了した場合にはバッファに保存され、ブロック i の処理が終了後、出力される。

【0067】この方法で、マルチパス・フェージング等による誤りのレベルに応じて必要最少限の訂正能力の誤り訂正符号を構成し、誤りを訂正することができる。また、バッファの大きさは $M \cdot (L - 1)$ ブロック分で良い。

【0068】本発明はまた R. A. Comroe, D. J. Costello らの方法と同様の方法に対しても適用できる。図8はこのように用いた本発明の他の実施例を示す図である。ここではRTD中に送信されるブロック数 M を、 $M = 2$ とする。図8において、1回目の送信で誤りが検出できなかったブロックは、そのままユーザに出力される。

【0069】ブロック i に誤りが検出されるとNAKが送信側に返され、先ず、ブロック i とそれに続くブロック $(i + 1)$ がバッファに保存される。そしてNAKを受けた送信側は2回目の送信でブロック i , $(i + 1)$ を送信する。ここで再送ブロック i , $(i + 1)$ はそれぞれ1回目に送信されたブロック i , $(i + 1)$ を情報ビットとする誤り訂正符号のパリティ・ビットからなる。これらの再送ブロックは、ブロック i に誤りが検出されなかった場合、パリティ・インバータによりブロック i が復元され、ユーザに出力される。またブロック i に誤りが検出された場合、バッファに保存されたブロック i と合わせて誤り訂正を行う。

【0070】それでもなお誤りが正せない場合は、バッファに保存されたブロック i を廃棄し、代わりに2回目に送信されたブロック i を保存する。同時にNAKを送信側に返し、送信側から3回目の再送ブロックとしてブロック i , $(i + 1)$ が送られる。

【0071】受信側では再送されたブロック i の誤り検出を行い、誤りがなければそのままユーザに出力し、誤りがあればバッファに保存されたブロック i と合わせて誤り訂正を行う。以下同様の要領で処理が繰り返される。

【0072】一方、ブロック $(i + 1)$ は受信後、ブロック i と全く同じ処理が行われる。そしてブロック i よりも先に処理が終了した場合にはバッファに保存され、ブロック i の処理が終了後、出力される。この間の処理は、図9(a)～(g)に i が3、4、5の場合で示した。

【0073】この方式によれば、受信側の持つバッファの大きさは M ブロック分で良い。

【0074】以上の実施例から分かるように、本発明では連続するブロックを再送するため、従来方式よりも簡単なアルゴリズムで再送、符号訂正などの処理を実現することができる。しかもバッファの容量が少なくてもオーバーフローすることがない。

【0075】図10に本発明の更に他の実施例のを示す。この方法は、SR方式とSAW方式を組み合わせたモード切り替え方式の改良にあたる。

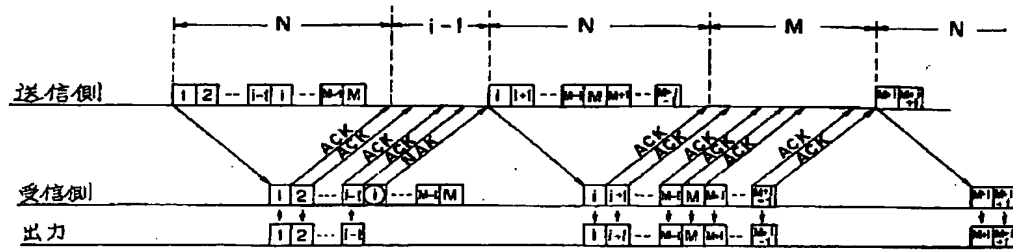
【0076】図10において、折り返し遅延時間(RTD)中に N 個のブロックが送信できるものとする、フレーム長 $M = kN$ ブロック (k は自然数) なるフレームを構成する。送信側はこの M 個のブロックを連続的に送信し、受信側では順次誤り検出を行い、ACK又はNAKを返送する。送信側ではNAKを受けたブロックのみを再送する。そうして M 個のブロック全てからACKが返るまでこの手順を繰り返し、 M 個のブロックの送信が完了すれば、次のフレームのブロックを送信する。この方法では、バッファのサイズは最大 kN ブロック分で良い。また、この k の値は通信路の状態に併せて決定し、通信路の状態がよく、誤りの少ない場合には k を大きく、通信路の状態が悪い場合には小さくする事で伝送効率を向上することができる。

【0077】図11は図10において $k = 2$ の場合について示した例である。図11で送信側からは $2N$ 個のブロックが順に連続的に送信され、受信側では受け取ったブロックについて順次誤り訂正を行い、送信側にブロック i 及び $2N$ に対してNAKが、そのほかのブロックに対してはACKが返ってきたとする。送信側では、NAKを受けるとそのブロックを直ぐに再送する。そしてブロック i 及び $2N$ に対して共にACKが送信側に返されるまで再送を続け、全てのブロックに対してACKが返ったら次の $2N$ 個のブロックを同じ手順で送信する。

【0078】図12は本発明の更に他の実施例を示す図である。図12は図11の場合と同じく、 $k = 2$ の場合について示している。図12において、送信側から $2N$ 個のブロックが順次連続的に送信され、受信側では受け取ったブロックについて順次誤り訂正を行い、ブロック i 及び $2N$ に対してNAK、そのほかのブロックに対してはACKを返したとする。送信側では、NAKを受けるとそのブロックを直ぐに再送する。ブロックを再送する方法として、図11では、ブロック i を送信してからブロック $2N$ を送信するまでの間、送信側から何も送信しない期間が続く、ブロック $2N$ を再送してからブロック i を再送するまで、再び空白期間が続くが、図12の例では、ブロック i を再送した後ブロック $2N$ を再送するまでの間、ブロック i を連続的に再送し続け、ブロック $2N$ を再送した後ブロック i を再送するまでの間はブロック $2N$ を連続的に再送し続ける。こうすることで再送回数を減らすことができ、伝送効率を更に向上することができる。

【0079】以上の図10～12の例では通信路の状態に合わせてフレーム長 M を変更出来るように考えたが、フレーム長 M だけでなく、1ブロックの長さを最適な値で伝送することにより、状態変化の激しい通信路において効率よく伝送を行うことができる。

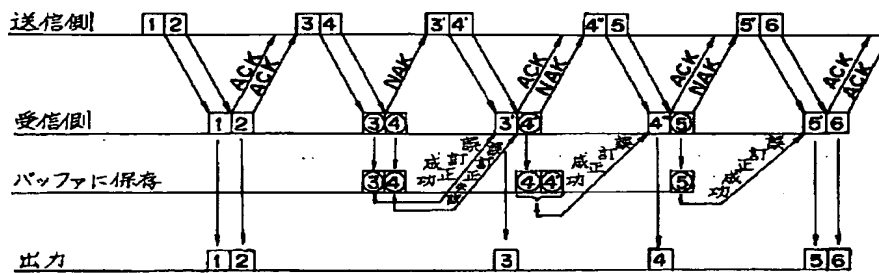
【図1】



① 受信側で誤りが検出されなかった i 番目ブロック

② 受信側で誤りが検出された i 番目ブロック

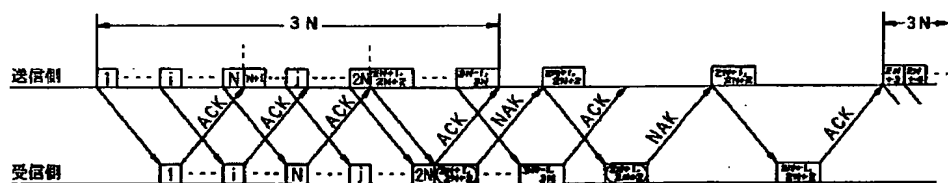
【図2】



① 受信側で誤りが検出されなかった i 番目ブロック

② 受信側で誤りが検出された i 番目ブロック

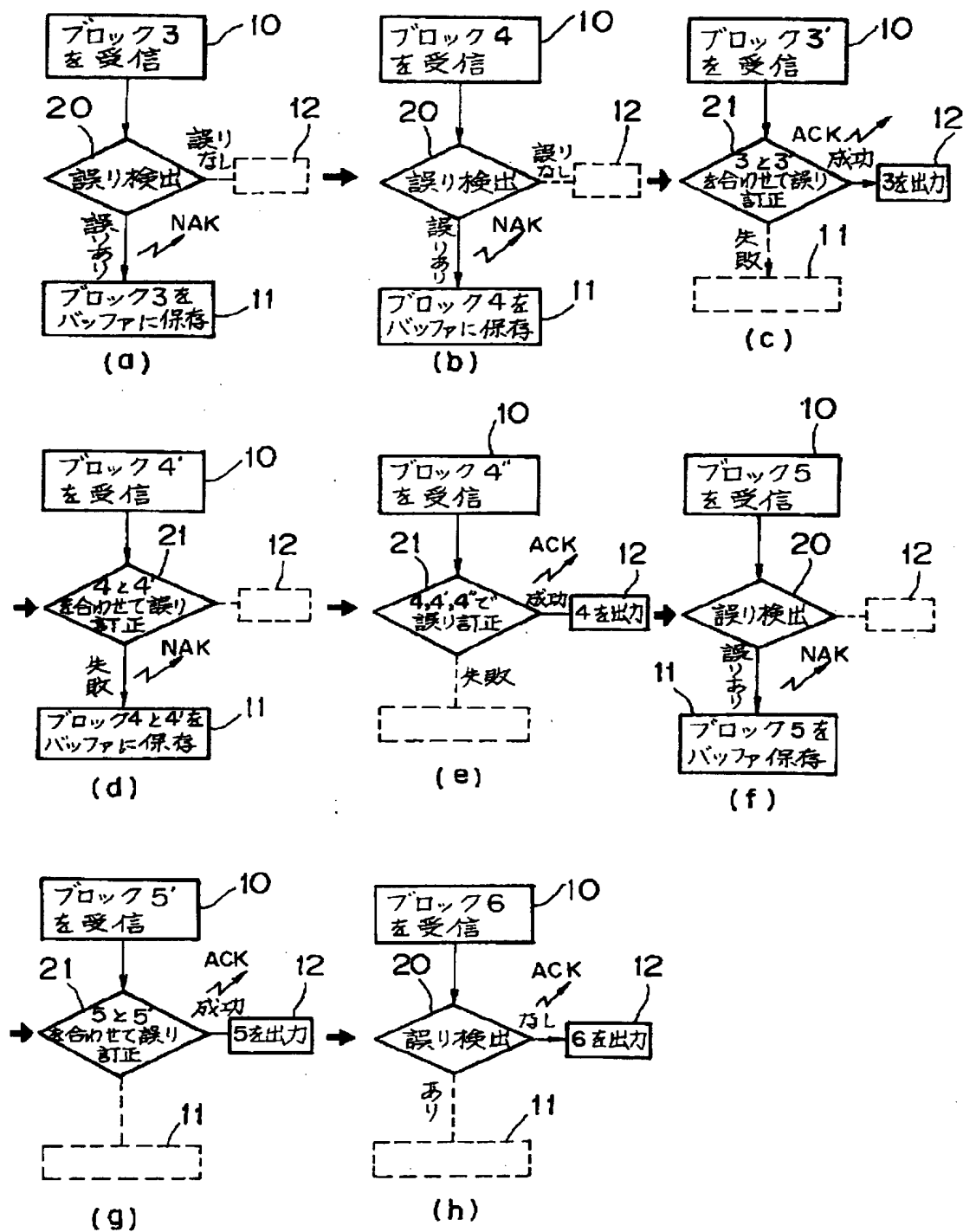
【図13】



① 受信側で誤りが検出されなかった i 番目ブロック

② 受信側で誤りが検出された i 番目ブロック

【図3】



The diagram illustrates the flow of data packets through four horizontal tracks:

- 送信側 (Sender Side):** Shows the sequence of packets being transmitted as pairs: (1, 2), (3, 4), (3', 4'), (4, 5), (5, 6).
- 受信側 (Receiver Side):** Shows the reception of packets. The first two are (1, 2) and (3, 4). After receiving (3', 4'), there is a period where no packet is received, followed by (4, 5) and (5, 6). Acknowledgments (ACK) are sent back to the sender after each successful reception.
- バッファに保存 (Buffer Storage):** This track shows how packets are stored in a buffer. Packets (3, 4) and (4, 5) are shown being moved into a buffer area labeled "バッファ". A note indicates "パケットは、バッファに格納される" (Packets are stored in the buffer).
- 出力 (Output):** This track shows the final output sequence of packets: (1, 2), (3), (4), (5, 6).
- 伝束 (Transmission Unit):** At the bottom, it shows a single unit containing packet (3).

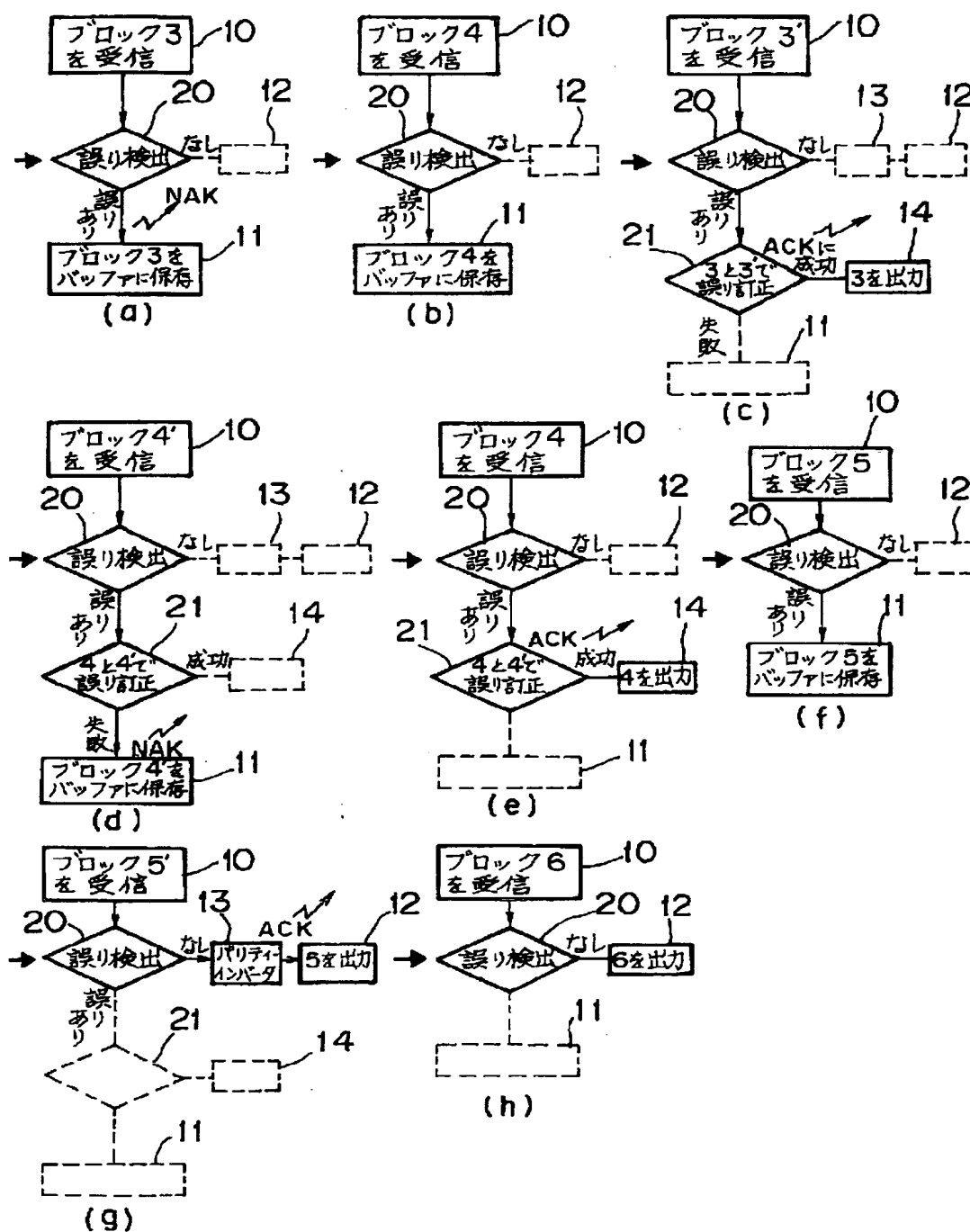
①受信側で誤りが検出されたi番目ブロック

Figure 1 is a sequence diagram illustrating the packet transmission system. The diagram is divided into two main horizontal sections: the top section for the transmitter (送信側) and the bottom section for the receiver (受信側). The transmitter track shows a sequence of packets: 1, 2, 3, 4, 3', 4', 5, 4'', 5', 6, 7. The receiver track shows the reception of these packets. Arrows indicate the flow of data from the transmitter to the receiver. Some packets are received out of order or are retransmitted, as indicated by the 'ACK' (Acknowledgment) and 'NAK' (Negative Acknowledgment) labels. A buffer section, labeled 'バッファに保存' (Stored in buffer), shows packets 3, 4, and 5 being stored and then output in order. The output section, labeled '出力' (Output), shows the final sequence of packets: 1, 2, 3, 4, 5, 6, 7.

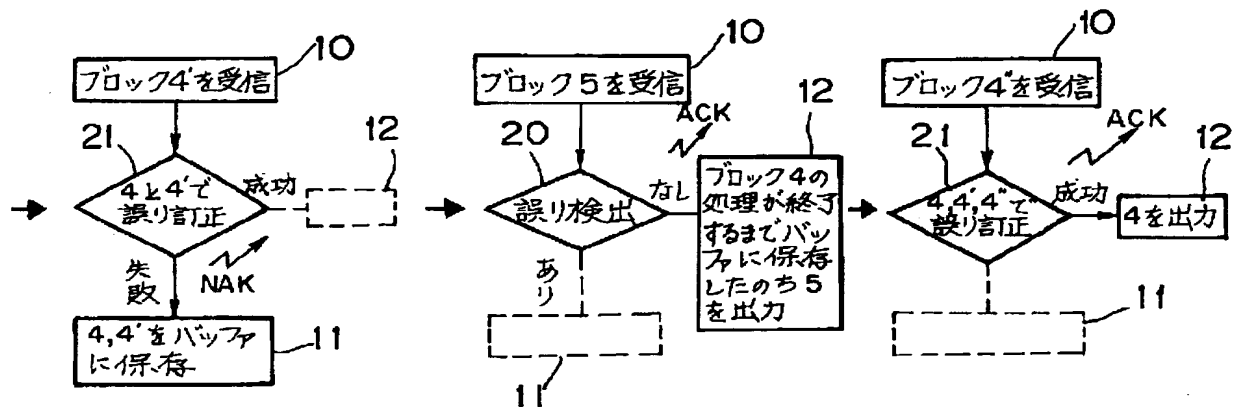
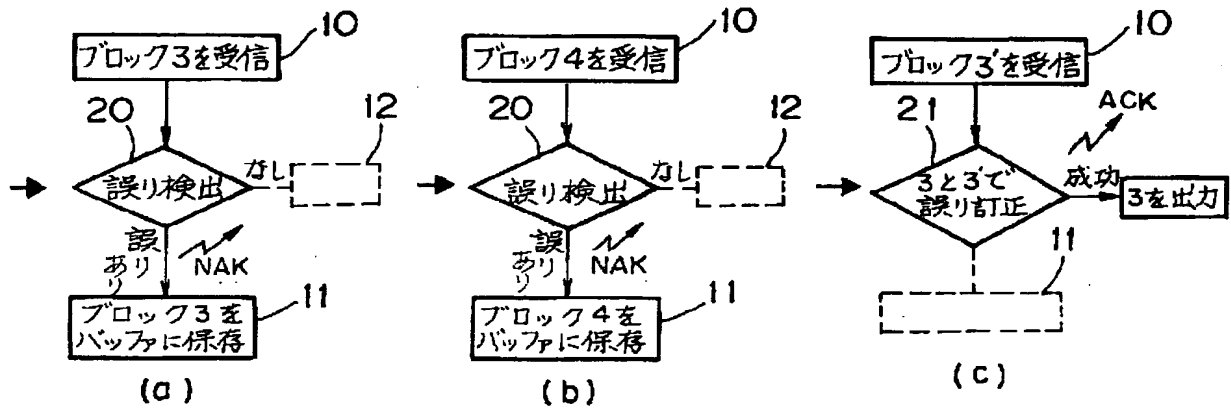
① 受信側で誤りが検出された1番目ブロック

❶ 受信側で誤りが検出された1番目ブロック

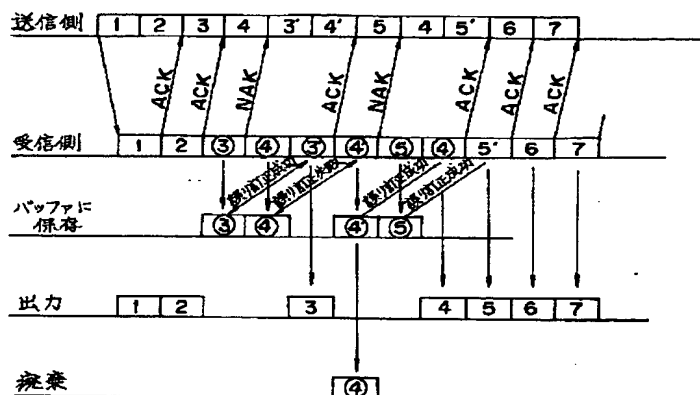
【図5】



【図7】



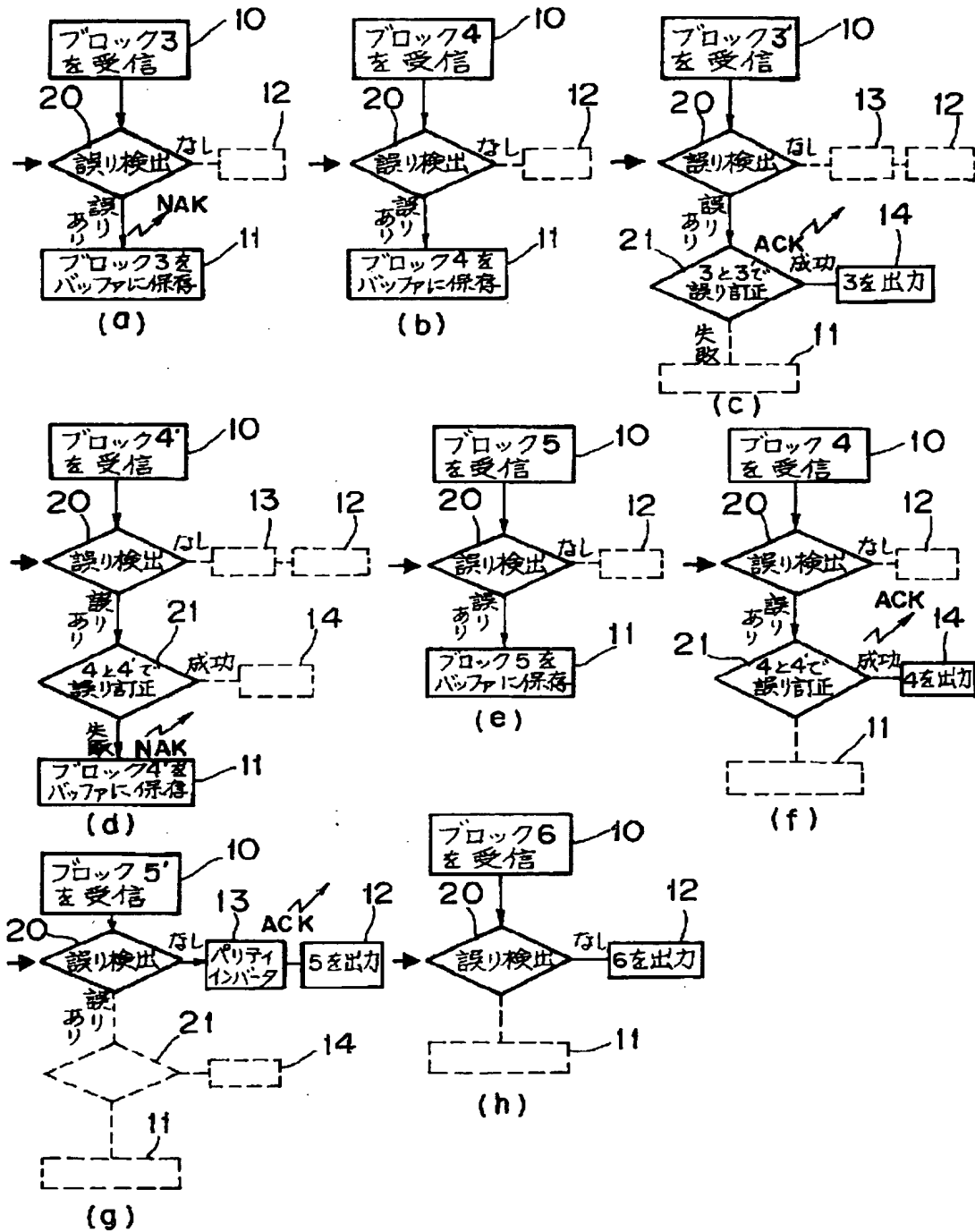
【図8】



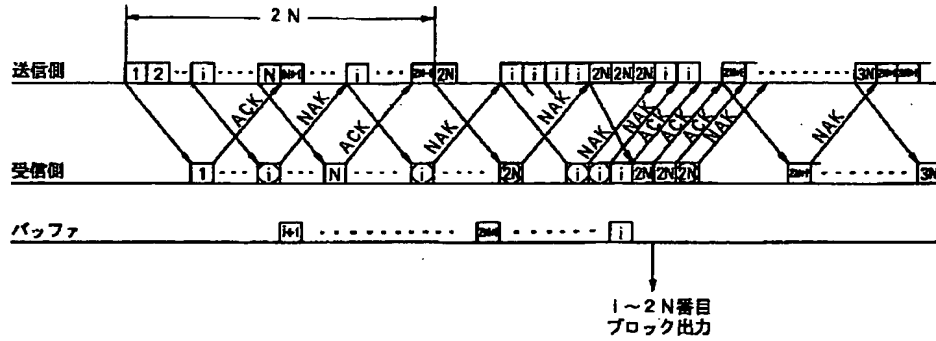
□ 受信側で誤りが検出されなかったi番目ブロック

① 受信側で誤りが検出されたi番目ブロック

【図9】



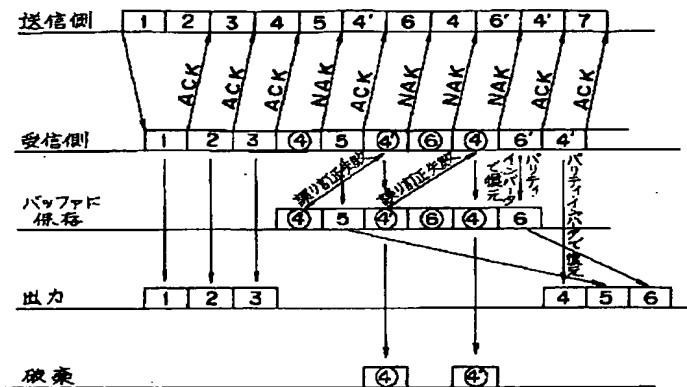
【図 12】



□ i 受信側で誤りが検出されなかった i 番目ブロック

□ i 受信側で誤りが検出された i 番目ブロック

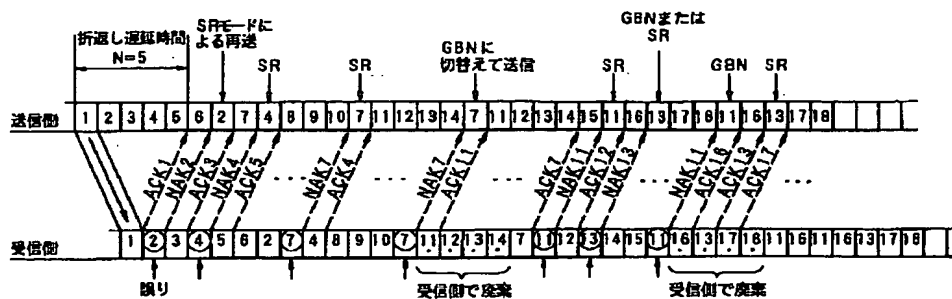
【図 14】



□ i 受信側で誤りが検出されなかった i 番目ブロック

□ i 受信側で誤りが検出された i 番目ブロック

【図 16】

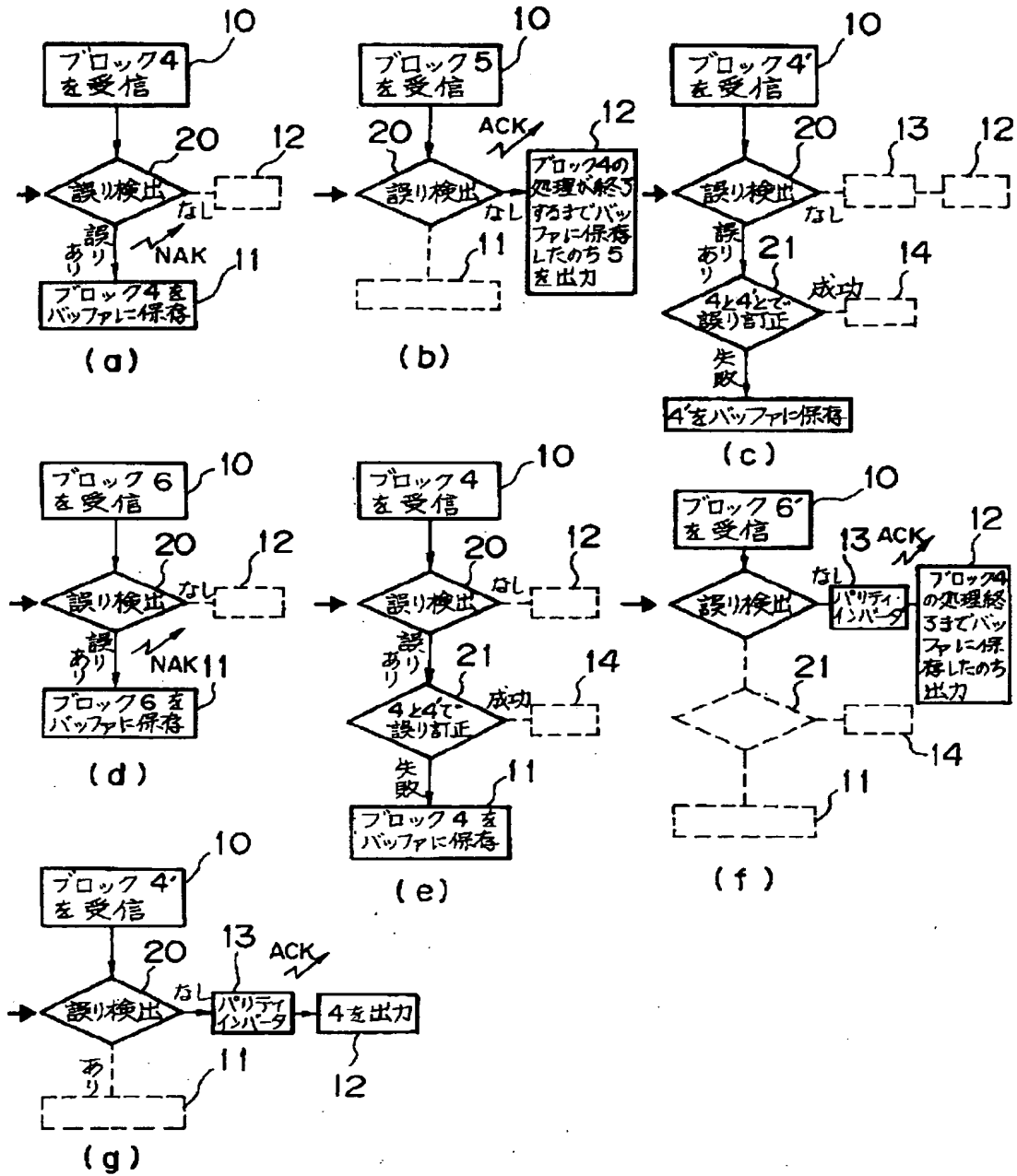


SRとGBNモードの切替え
v=1 N=5

□ i 受信側で誤りが検出されなかった i 番目ブロック

□ i 受信側で誤りが検出された i 番目ブロック

【図15】



【図17】

